Floating IP range

(See also: Difference_between_Floating_IP_and_private_IP)

OpenStack instances receive a private IP address through which they can reach each other and through which hosts can reach them. In order to access these instances from other machines in your network, such as your workstation, the instances will need to be allocated a "floating IP." Packstack automatically configures this with a default that may well be wrong for your network. You'll want to configure your OpenStack install with a range of free IP addresses that's correct for your network.

If you don't know of an appropriate range of IP addresses on your network (best), and can't ask someone who does know (next best), you can make an intelligent guess by steering well clear of the range you typically get DHCP addresses in, by picking a fairly small range (/29 gives an 8 address range, 6 of which will be usable), and by using nmap to check if hosts are up in the range you're guessing at.

For instance, 192.168.1.56/29 represents a small range of addresses (192.168.1.56-63, with 57-62 usable), and you could run the command "nmap -sn 192.168.1.56/29" to check and see if that whole range was in fact unused (at the moment, at least).

Steps to remove hard-coded floating IP and add a new one:

With nova-network:

1. `source /root/keystonerc_admin`
2. `nova floating-ip-bulk-delete 10.3.4.0/22` (this is the hard-coded range in packstack)
3. `nova floating-ip-bulk-create CORRECT-RANGE` (in the above example, this would be 192.168.1.56/29)
4. `nova-manage floating list` (to see your new list of floating IPs)

With Neutron:

No floating IPs are created by default. You must first create a public network and subnet, defining the IP address range for floating IPs at that time.

1. `source /root/keystonerc_admin`
2. `neutron net-create public --router:external`
3. `neutron subnet-create public 192.168.1.0/24 --name vlan --enable_dhcp=False --allocation_pool start=192.168.1.57,end=192.168.1.62 --gateway 192.168.1.1` (use your network gateway here - change the IP addresses in the allocation range to match what is available on your network)
4. `neutron router-create router1` (router1 has to be replaced by the name of your router)
5. `neutron router-gateway-set $router_id $vlan_id` (use your router id and previous created vlan id)
6. `neutron floatingip-create public` (repeat as necessary)

Neutron does not auto-create floating IPs or auto-assign them to new instances but this feature is planned for future inclusion.